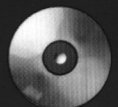
USB COMPLETE

Everything You Need to Develop Custom USB Peripherals

No custom drivers needed—use the Win32 API and Visual Basic

JAN AXELSON

author of *Parallel Port Complete* and *Serial Port Complete*



INCLUDES CD-ROM

Table of Contents

Introduction xiii

1. A Fresh Start 1

What USB Can Do 3
User Benefits
Developer Advantages
It's Not Perfect... 11
User Challenges
Developer Challenges
History 15
The Motivation for Change
The Specification's Release
USB 2.0
USB and IEEE-1394

2. Is USB Right for My Project? 19

Fast Facts 19
Minimum PC Requirements
The Components

Bus Topology
Defining Terms
Ports
The Host's Duties
The Peripheral's Duties
Speed
The Development Process 31
Elements in the Link
Tools
Steps in Developing a Project

3. Inside USB Transfers 35

Transfer Basics 36
Configuration Communications
Application Communications
Managing Data on the Bus
PC-to-PC Communications
Elements of a Transfer 39
Device Endpoints
Pipes: Connecting Endpoints to the Host
Types of Transfers
Stream and Message Pipes
Initiating a Transfer
Transactions: the Building Blocks of a Transfer
Phases of a Transaction
Ensuring that Transfers Are Successful 50
Handshaking
Reporting the Status of Control Transfers
Error checking

4. A Transfer Type for Every Purpose 57

Control Transfers 57
Availability
Structure
Data Size
Speed
Detecting and Handling Errors
Bulk Transfers 63
Availability
Structure
Data Size
Speed

Detecting and Handling Errors
Interrupt Transfers 66
Availability
Structure
Data Size
Speed
Detecting and Handling Errors
Isochronous Transfers 69
Availability
Structure
Data Size
Speed
Detecting and Handling Errors

5. Enumeration: How the Host Learns about Devices 73

The Process 74
Enumeration Steps
Enumerating a Hub
Device Removal
About Descriptors 80
Types
Device Descriptor
Configuration Descriptor
Interface Descriptor
Endpoint Descriptor
String Descriptor

6. Control Transfers: Structured Requests for Critical Data 91

Elements of a Control Transfer 91
The Setup Stage
The Data Stage
The Status Stage
Handling Errors
The 11 Standard Requests 99
Set_Address
Get_Descriptor
Set_Descriptor
Set_Configuration
Get_Configuration
Set Interface
Get_Interface

Set_Feature
Clear_Feature
Get_Status
Synch_Frame
Other Control Requests 112
Class-Specific Requests
Device-Specific Requests

7. Chip Choices 115

Elements of a USB Controller 116
CPU
Program Memory
Data Memory
Registers
USB Port
USB Buffers
Other I/O
Other Features
Simplifying the Development Process 121
Architecture Choices
Chip Documentation
Sample Firmware
Driver Choices
Debugging Tools
Project Needs
A Look at Some Chips 129
Intel/Cypress 8x931 and 8x930
Cypress Semiconductor EZ-USB
Cypress Semiconductor CY7C63X00
NetChip NET2888
National Semiconductor USBN9602
Philips Semiconductors PDIUSBD11/12

8. Inside a USB Controller: the Cypress CY7C63001 141

Selecting a Chip 142
Requirements
The Choice
Inside the Chip 143
Features and Limits
Architecture
Memory
I/O

Interrupt Processing
The Timer
Resets
Power Management
The Serial Interface Engine (SIE)
Hardware and Firmware Responsibilities

9. Writing Firmware: the Cypress CY7C63001 169

The Assembler 169
Assembly Programming Basics
Assembler Codes
Using the Assembler
If You Prefer C... 175
Advantages to C
Using the Compiler
Hardware Development Tools 177
The Developer's Kit
The Starter Kit
EPROM Programming

10. How the Host Communicates 191

Device Driver Basics 191
Insulating Applications from the Details
Options for USB Devices
How Applications Communicate with Devices
The Win32 Driver Model 197
Driver Models for Different Windows Flavors
Layered Drivers
The Communications Flow
More Examples
Choosing a Driver Type 207
Drivers Included with Windows
Vendor-supplied Drivers
Other Generic Drivers
Custom Drivers
Writing a Custom Driver 211
Requirements
Using a Driver Toolkit

11. How Windows Selects a Driver 217

The Process 217
The Driver Information Database

The Registry's Role
The Control Panel
What the User Sees
Tips for Experimenting
Inside an INF File 226
Sections
The Generic INF File for HIDs

12. Human Interface Devices: Firmware Basics 235

What is a HID? 236
Hardware Requirements
Firmware Requirements
Identifying a Device as a HID 240
Descriptor Contents
HID Class Descriptor
Report Descriptors
HID-specific Requests 248
Get_Report
Set_Report
Get_Idle
Set_Idle
Get_Protocol
Set_Protocol
Transferring Data 256
Sending Data to the Host
Receiving Data from the Host

13. Human Interface Devices: Reports 263

Report Structure 263
Using the HID Descriptor Tool
Predefined Values
Short Items
Long Items
The Main Item Type 267
Input, Output, and Feature Items
Collection and End Collection Tags
The Global Item Type 272
Identifying the Report
Describing the Data's Use
Converting Raw Data
Describing the Data's Size and Format
Saving and Restoring Global Items

The Local Item Type 280
Physical Descriptors
Padding

14. Human Interface Devices: Host Applications 285
Host Communications Overview 285
How the Host Finds a Device
Documentation
Calling API Functions from Visual Basic 287
The Declaration
Calling API Functions
Two Useful Routines
Finding a Device 298
Obtain the GUID for all HIDs
Get an Array of Structures with Information about the HIDs
Identify Each HID Interface
Get the Device Pathname
Get a Handle for the Device
Read the Vendor and Product IDs
Get a Pointer to a Buffer with Device Capabilities
Get the Device's Capabilities
Get the Capabilities of the Values
Reading and Writing Data 312
Write Data to the Device
Read Data from the Device
Closing Communications 316

15. Device Testing 319
USBCheck 320
Detecting a Device
The Tests
HIDView
Protocol Analyzers 324
Hardware
Software
Implementers Forum Resources 331
The Compliance Program
Testing Opportunities

16. Hubs: the Link between Devices and the Host 337
Hub Basics 338
The Repeater

Managing Communications
Endpoints
Speed
How Many Hubs in Series?
The Hub Class 343
Hub Values for the Standard Descriptors
The Hub Descriptor
Hub-class Requests

17. Managing Power 347

Powering Options 347
Voltages
Which Peripherals Can Use Bus Power?
Power Needs
Informing the Host
Hub Power 352
Power Sources
Over-current Protection
Power Switching
Saving Power 354
Global and Selective Suspends
Current Limits for Suspended Devices
Resuming Communications

18. The Electrical Interface 357

Signal Voltages 358
Differential 1 and 0
Other Valid States
Cables 360
Conductors
Connectors
Detachable and Captive Cables
Cable Length
Ensuring Signal Quality 366
Sources of Noise
Balanced Lines
Twisted Pairs
Shielding
Edge Rates

19. Signals and Encoding 371

Bus States 371

J and K
Resume
Connect
Start-of-Packet
End-of-Packet State
Disconnect State
Reset State
Data Encoding 373
Staying Synchronized
Timing Accuracy
Packet Format 377
Sync Field
Packet Identifier Field
Inter-packet Delay

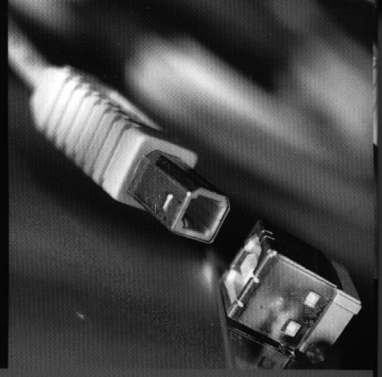
Appendix A. Resources 381

Appendix B. Cypress CY7C63001 Registers 383

Index 387

USB COMPLETE

JAN AXELSON



Put the power of USB in your designs

The Universal Serial Bus was designed from the ground up to provide a single, easy-to-use interface for computer peripherals. But the price of simplicity for users is greater complexity for the developers who design and program USB devices.

In this hands-on guide, Jan takes the pain out of designing for USB. Find the answers to all of your questions:

- Can my project use a USB interface?
- Which peripheral-controller chip should my design use?
- How do I access USB peripherals from Visual-Basic applications?
- What firmware code does my peripheral need in order to communicate with PCs?
- Can my design use bus power or does it need its own supply?

Plus—learn about transfer types, interfacing to human-interface devices (HIDs), USB signals and encoding, and much more!

"Many books are full of things that are easy to find out, and skirt around the harder stuff, which you have to really work at. What I really like about this book is that Jan has obviously slogged at the difficult stuff as well."

Dave Wright
Staff Applications Engineer
Cypress Semiconductor

"I tell all my students that they really need this book in their library."

Paul E. Berg, instructor
Annabooks USB Developers Workshop

"USB Complete *provides a great groundwork for anyone working with USB for any purpose, whether it is designing a peripheral or creating host software. This book should be read by anybody getting started with USB."*

Joshua Buergel
BlueWater Systems

Jan Axelson is the author of *Parallel Port Complete*, *Serial Port Complete*, *The Microcontroller Idea Book*, and *Making Printed Circuit Boards*. Bill Machrone of *PC Magazine* has described Jan's writing as "a model of clarity and conciseness."

Published by
Lakeview Research
www.lvr.com

Shelve under
Computer Hardware

Price $49.95 / ISBN 0-9650819-3-1
ISBN 096508193-1
9 780965 081931 54995